AP9406V1

An Evaluation of OASIS-CC for use in the FOS of the ECS Project

Working Paper

June 1994

Prepared Under Contract NAS5-60000

RESPONSIBLE ENGINEER

John R. Kuntz /s/ 6/10/94

Jon R. Kuntz, FOS Supervisor — Date
EOSDIS Core System Project

SUBMITTED BY

T. Anders /s/ 6/10/94

Tony Anders, FOS Manager — Date
EOSDIS Core System Project

Hughes Applied Information Systems
Landover, Maryland

This page intentionally left blank.

Contents

1. Introduction

1.1 Purpose 1
1.2 Scope 1
1.3 Organization 1
1.4 Review and Approval 2

2. Overview of OASIS-CC

2.1 Data Handling 3
2.2 Command 3
2.3 External Communications 4
2.4 Bridge 4
2.5 User Interface 5
2.5.1 CSTOL 5
2.5.2 Display 5
2.6 File Outputs 5
2.7 Database 6

3. Evaluation Process

3.1 Evaluation Criteria 7
3.2 Evaluation Approach 7
3.2.1 Documentation Review 7
3.2.2 Hands-On Evaluation 8
3.2.3 Training Course 8
3.2.4 OASIS Working Group 8
3.2.5 Additional Activities 9

4. Evaluation Results

4.1 General Comments 10
4.2 FOS Specific Results 11
4.2.1 Functional and Performance Requirements 12
4.2.2 Flight Operational Requirements 14
4.2.3 Integration into the FOS 15
4.2.4 Additional Functionality/Evolvability 15
4.2.5 Changes in Industry Standards 16
4.3 Summary 16

5. Conclusions

5.1 Recommendations 17

Abbreviations and Acronyms

1. Introduction

1.1 Purpose

The purpose of the OASIS evaluation was to examine the feasibility of integrating the Operations and Science Instrument Support - Command and Control (OASIS-CC or OASIS) software package into the EOSDIS Core System (ECS) Flight Operations Segment (FOS). OASIS was developed by the University of Colorado and has been selected as the Integration and Test (I&T) tool to be used by the spacecraft and instrument providers for both the Earth Observing System (EOS) AM1 and PM1 missions.

1.2 Scope

The potential use of OASIS needs to be evaluated with respect to the requirements imposed upon the FOS operational system. The FOS must be capable of supporting a multi-mission environment in which it must provide simultaneous real-time and off-line support for multiple spacecraft and their instruments. The requirements placed upon the FOS dictate that it must be an integrated system that provides the resources to plan and schedule the spacecraft and instrument activities; generates memory loads for the spacecraft and instruments which contain stored commands, table updates, and flight software updates; responds to requests for changes to the agreed schedules; handles multiple asynchronous interfaces; supports real-time contacts with the spacecraft; provides analysis capabilities for both routine and anomalous conditions; and in general assists the Flight Operations Team (FOT) with the decision making process required to maintain the health and safety of the spacecraft and its instrument payload.

An I&T environment, which OASIS is currently supporting for the AM1 mission and will be supporting for the PM1 mission, has very different requirements. OASIS is being used as a single user system for the purpose of testing the spacecraft or instrument in a controlled laboratory environment. This includes the sending of commands to the spacecraft or instrument and processing the resulting telemetry. OASIS must provide user defined displays and a database for defining the spacecraft or instrument. Due to the many differences between the I&T laboratory environment and an operational environment, a careful examination of OASIS was needed to evaluate the benefits and/or risks associated with integrating OASIS into the FOS. The scope of this study was therefore to evaluate the ability of OASIS to satisfy the FOS requirements and to be integrated into a complete FOS system.

1.3 Organization

This paper is organized as follows:

Section 2.0 provides an overview of OASIS. Included within this section are descriptions of the OASIS subsystems. Section 3.0 provides a description of the evaluation process, which includes the evaluation criteria and approach. Section 4.0 presents the results of the evaluation. The recommendations which resulted from the evaluation are presented in Section 5.0.

1.4 Review and Approval

This document is an informal contract deliverable approved at the Office Manager level. It does not require formal Government review or approval; however, it is submitted with the intent that review and comments will be forthcoming.

The ideas expressed in this White Paper are valid for June 1994; the concepts presented here are expected to migrate into the following formal CDRL deliveries:

Table 1-1. White Paper to CDRL Migration

White Paper Section	CDRL DID/Document Number	Document Name
5.0	304/DV1	Segment/Element Requirements Specification
5.0	305/DV2	Element Design Specifications
5.0	604/OP1	ECS Operations Concept Document

Questions regarding technical information contained within this Paper should be addressed to the following ECS and/or GSFC contacts:

- ECS Contacts
  - Jon R. Kuntz, FOS Supervisor, 301-925-0622, jkuntz@eos.hitc.com
- GSFC Contacts
  - Mike Rackley, FOS Real-Time System Manager, 301-286-2220, mrackley@gsfcmail.nasa.gov

2. Overview of OASIS-CC

OASIS was developed by the University of Colorado at Boulder in the Laboratory for Atmospheric and Space Physics (LASP). OASIS was developed to provide the resources necessary to monitor and control many of the functions of a spacecraft and its scientific instruments. OASIS is able to receive telemetry from a spacecraft or instrument and extract the individual data items from the telemetry stream. Additionally, OASIS is able to accept commands from a user and translate them into either binary or ASCII format for transmission to the spacecraft or instrument. For display purposes, OASIS allows the user to define and save display formats to be used when required.

OASIS is composed of several subsystems which are capable of communicating with one another. Communication is also possible with an external system and the user. The following provides a brief description of the components of OASIS.

2.1 Data Handling

The Data Handling subsystem is responsible for processing and converting telemetry according to definitions provided in the database tables. Incoming data can come from a spacecraft, an instrument, or computer. The telemetry data can be displayed or stored.

The Data Handling subsystem keys off the latest data table. The latest data table defines all the attributes of each data item to be processed and is the repository for the most recent value of each data item. The user can define in the latest data table any special processing to be performed on each data item. The special processing would include state conversions, engineering unit conversions, limit checking, triggers, and equation processing.

2.2 Command

The Command subsystem is responsible for translating user commands into the format expected by the instrument or spacecraft and supplying the formatted commands to the External Communications subsystem for transmission. OASIS commands are issued in Colorado System Test and Operations Language (CSTOL) format. Various commanding database tables are used to translate the CSTOL format commands into the bit-string format recognized by the spacecraft simulator or instrument for which the command was intended.

The commands are separated into three broad categories: immediate (processed upon receipt), timed (processed at the requested time) and priority (sent during the assembly of a command message buffer). These are assembled into their corresponding message types: real-time, delayed, and priority.

Command validation is supported by defining a command as hazardous in the data base, which requires the operator to validate it before it can be transmitted.

OASIS commanding supports commands in both ASCII (variable length string segments) and binary (fixed length) formats.

The supported levels of command checking include prechecking (prerequisite state checking), acceptance checking (command transmission) and postchecking (end item verification).

Security for the OASIS command subsystem is provided through the use of a USER_PRIVILEGES table which restricts user access to the system.

2.3 External Communications

The primary function of the External Communications subsystem is to provide OASIS the capability of transmitting commands to the spacecraft or instrument. It is also responsible for accepting data packets or frames from external systems and making that data available to the Data Handling subsystem for further processing. A system of four (4) database tables are used to manage the information required to perform interfacing. These tables are labeled "streams", "links", "decomposition", and "latest_data". The interfacing information contained in the tables includes communication link, sending / receiving protocol, and data extraction methods. External communications protocols supported include Internet, TDM (time division multiplexed) and CCSDS packetization. In addition VMS and UNIX formatted data is supported.

The streams are divided into three main categories: primary, secondary and substreams. A stream, depending on the type, may be acquired, become decomposed, or be (re)routed. All data streams must have a primary definition that specifies either return (telemetry) or forward (commanding). The stream may also be defined as a secondary stream. This allows the stream to be forwarded to an external process, or to a Control Language Processor (CLP) within OASIS. The substream category is provided for OASIS internal processing, and will either contain telemetry data, or be processed automatically as per database defined entry, depending upon its substream definition.

2.4 Bridge

The OASIS bridge feature allows for the routing of data to a file or to another process, and is the mechanism by which telemetry data values are archived. It accommodates both binary and ASCII data. Each bridge requires corresponding entries in the database.

To establish a bridge, the bridge processor name must be entered into the database. For communications bridges (i.e., links with other processes), the stream processor, a link record name and associated information must be defined in the database. A bridge that is output to a file will result in three (3) files: header file, id file, and a data file.

The bridge output data is defined in the database by specifying the item (parameter name) and format (raw, converted, smoothed or trend). The bridge output options include output of time, converted units, mode (keyed or non-keyed; keyed indicates the parameter that is to trigger the bridge output), frequency specification (each occurrence, or only upon a change in value) and output of quality flag.

2.5 User Interface

The User Interface is comprised of a combined graphical point-and-click and command line interface. The User Interface is built using the Transportable Applications Environment (TAE+) software development environment. It conforms to industry standards for graphical user interface design utilizing MIT's X-window system X11R4 and OSF Motif toolkit. To tailor TAE+ to the specific needs of OASIS, specialized modifications were required of TAE+. TAE+ provides a workbench as a development tool that supports the design and layout of an application's interface. The workbench provides all the basic tools that a developer would need to build an interface including the capability to automatically generate code for the interface. The User Interface is composed of the CSTOL command language and the Display subsystem.

2.5.1 CSTOL

OASIS uses the CSTOL system control language. CSTOL is a command language used to provide functions to monitor and control the spacecraft, its instruments and their ground support equipment. CSTOL is an enhanced version of Goddard Space Flight Center's (GSFC) System Test and Operations Language (STOL) and has an English-like syntax. It provides a mechanism for expanding the language through macros and the ability to access database tables using a query language. It also provides the capability to define and execute procedures.

2.5.2 Display

The Display subsystem provides a method for the real-time display of data. Data items which are to be displayed must be defined in the database tables. When OASIS receives new values it updates the database with the new values. Values can be raw, converted to engineering units, smoothed to present the average of the converted values, or trended to represent the item's rate of change. Any of these values may be examined or displayed by the user.

OASIS employs its own special windows. These windows are defined by the OASIS database. The Motif window manager provides for any of these windows to be moved or resized. Default location values can be defined as well. These windows are : CSTOL Prompt, CSTOL Error, CSTOL Procedure, CSTOL Ask, CSTOL Report, Command Window, Message Window and Alert Window.

2.6 File Outputs

The user has the capability of generating several types of output files. All output files are created by issuing a CSTOL directive.

The different types of output files are:

- Message logs which contain system activities, events and error messages.
- Raw telemetry data files which are recorded so that the user can perform playbacks.
- Bridge files which allow the user to request subsets of telemetry data be sent to a file, or to an external system.

- Procedure listings which can be sent to a window or file.
- Database report which contains a dump of a selected database file.
- Snaps which allow for the contents of a telemetry display page be dumped to a file.

2.7 Database

The OASIS database contains the spacecraft or instrument specific definitions required for commanding, monitoring and display of the spacecraft or instrument data. OASIS allows for tailoring of the database tables to fit the users application. User and system access to the database tables is provided.

OASIS reads the database tables into memory when it is initialized. Once OASIS has been initialized the user can read or modify (i.e., insert, delete or update) records in the database. Any database modifications made while OASIS is running only affect the memory resident version of the tables.

3. Evaluation Process

3.1 Evaluation Criteria

Throughout the evaluation process the following list of criteria were used to determine the feasibility of using OASIS within the FOS:

- Ability of OASIS to meet the FOS functional and performance requirements.
- Ability of OASIS to support the flight operational requirements.
- Ability to integrate OASIS into the rest of the FOS.
- Ability to add functionality to OASIS.
- Ability of OASIS to evolve as required by the dynamics of the EOS mission.
- Ability of OASIS to adapt to changes in the industry standards.

3.2 Evaluation Approach

The evaluation of OASIS was conducted in a manner that would provide the best possible exposure to OASIS. An evaluation team was assembled which included personnel with extensive experience in the various components of satellite control centers. Control center experience included real-time command and control, user interface, database design and utilization, and off-line processing. Additional control center personnel, including operations personnel, were consulted on an as needed basis. Every attempt was made to utilize all possible avenues to gain a better insight and understanding of the capabilities of OASIS.

3.2.1 Documentation Review

The first step in evaluating the OASIS product was to secure and review the full set of available documents on OASIS. The documents were received from the NASA library and made available to the evaluation team. The team reviewed all the documents and paid particular attention to those sections within the various documents which would be relevant to their particular expertise. The list of the documents received and reviewed is as follows:

- OASIS-CC System Manager's Guide, V02.05.08, June 1993
- OASIS-CC Quick Reference Manual, V02.05.08, June 1993
- OASIS-CC Application Environment Reference Manual - UNIX Version, Copyright 1992
- OASIS Database Builder (DBB) - Release 1.0, User's Guide, June 1993
- OASIS-CC CSTOL Reference Manual, V02.05.09, February 1994
- OASIS-CC System Manager's Guide, V02.05.10, February 1994

3.2.2 Hands-On Evaluation

The hands-on evaluation of the OASIS product was performed over a 10 week period of time. In order to focus on the OASIS capabilities rather than developing custom software to interface with OASIS, the *spectrometer* program delivered with OASIS was used for the evaluation. The *spectrometer* is capable of generating housekeeping and science telemetry. It is able to receive commands and modify its telemetry according to the command information.

For purposes of the evaluation a stand-alone SUN SPARCstation LX was configured as follows:

- 24 MegaBytes Memory
- 424 MegaBytes Disk Storage
- 1/4" Tapedrive
- SunOS 4.1.3C
- X11R4
- Motif 1.1.4
- TAE+ 5.2
- OASIS Version 8

A copy of OASIS-CC version 10 was received during the final stages of the hands-on evaluation, therefore the final 2 weeks of the evaluation period were dedicated to performing an evaluation on the new version of OASIS. This allowed further verification of results obtained from version 8 and the evaluation of the additional capabilities provided in the version 10 release.

3.2.3 Training Course

Several members of the evaluation team attended the training course offered by the University of Colorado on OASIS. FOS attendees were present at the training course offered in Boulder, Colorado in June 1993 and the training course offered in Greenbelt, Maryland in March 1994. Since the two training courses offered the same information the attendance at the two sessions was staggered to allow the maximum number of people to be trained by the University of Colorado.

3.2.4 OASIS Working Group

In an effort to interact with and learn from other users of OASIS, the evaluation team participated in the OASIS Working Group meetings that were held at Goddard Space Flight Center in August 1993 and March 1994.

3.2.5 Additional Activities

When appropriate, additional means of gathering relevant OASIS information were used. The following list references the additional sources of OASIS information which were sought out and synthesized into the evaluation process.

- Communications with Fred Lacey at the University of Colorado, LASP: Numerous telephone calls and e-mail correspondences were sent to Mr. Lacey. This was particularly true during the installation, early portions of the hands-on evaluation, and when questions or problems arose which couldn't be answered locally. Mr. Lacey's insight and assistance was invaluable during the evaluation process.
- Attendance at design reviews: To gain insight into how OASIS is being used within the I&T environment, formal reviews sponsored by Martin Marietta Corporation (MMC) were attended. These reviews included the Flight Software Testbed PDR and the Spacecraft Simulator PDR.
- Demonstration by MMC: MMC provided a demonstration of their system for I&T and the role of OASIS within that system. MMC also provided "lessons learned" information regarding OASIS.
- Tutorial by Jeff Bowser of Hughes STX. Mr. Bowser is responsible for the development of the OASIS Database Builder. Mr. Bowser provided much information, particularly on the installation and startup of OASIS. He also spent time at the ECS facility giving a tutorial on OASIS.

4. Evaluation Results

4.1 General Comments

OASIS was found to be very effective and therefore a valuable tool when used within certain constraints. It is quite evident why OASIS has been chosen by the EOS AM1 spacecraft and instrument providers as their command, monitoring, and display system for spacecraft/instrument integration and test. OASIS is useful as a single user system that allows the user to monitor an instrument, build and send commands, and create a wide variety of customized displays.

The data handling subsystem allows engineering unit conversion from raw counts by either a linear conversion or a polynomial conversion up to and including a 3^{rd} order polynomial. Both red and yellow limits are supported as are delta limit checks and trend limits. An excellent feature of the system is the use of triggers which provide an automatic reaction to a defined limit violation. The trigger, when activated by a limit violation, will start a procedure which performs a predefined task. Trend limits were also found to be a unique feature of the OASIS system. Trend limits are used to calculate the amount of time it will take a data item to reach a limit violation if it continues changing at the current rate. OASIS also supports pseudo-telemetry by allowing the user to define equations which define a new data point and use existing telemetry values as input. The equations are calculated, thus resulting in an updated value for the pseudo-telemetry point, when a new value for an input point is received.

The command subsystem provides the capability to generate and transmit both real-time and absolute time commands. The real-time commands are sent immediately upon input and are to be executed by the spacecraft/instrument upon receipt. Timed commands are time tagged with an execution time and are stored in the on-board processor for execution at the designated time. OASIS does allow the user to build a stored command load file which contains multiple commands. These stored commands are time tagged with an absolute time. OASIS provides the capability to define a command as hazardous in which case the user must approve the command prior to it being transmitted. If a command is defined as safe, it is sent without having to be approved prior to transmission. Command validation and execution verification can be accomplished through the use of procedures, which when executed perform the required task.

The display subsystem, which is based upon the GSFC developed TAE+ product, provides a workbench capable of creating a wide variety of user specified displays. A feature which all users look for in a system is the ability to customize their displays and save them in the database so they will always be available. OASIS provides this capability, thus increasing efficiency and consistency when using the system. TAE+ provides a wide assortment of widgets which are available for screen customization. OASIS allows any item which is defined in the latest_value_table to be displayed.

Additionally, OASIS provides many other capabilities which are both useful and required. OASIS allows the user the capability to generate message logs, database reports, raw telemetry files, bridge files, procedure listings, and snap files. Within an I&T environment the ability to modify the database easily is a necessity. OASIS provides this capability which allows testing to continue without extended time lost while generating a new database. Another important feature is being able to route data, whether it be to another process, a file, or even an external system. The bridge feature within OASIS provides this capability and furthermore, the information for the definition of a bridge is contained in the database so it is always accessible by the user.

Overall, when used in the proper context OASIS was found to be a very valuable tool. It provides the basic capabilities of the traditional real-time control center along with a few added features such as triggers and trend limits which expand the capabilities of the control center. OASIS seems to be particularly well suited for the spacecraft and/or instrument I&T environment.

4.2 FOS Specific Results

The current operational concept for the FOS is to allow the FOT the capability to monitor any combination of spacecraft and instruments from a single operator position. One FOT member might want to monitor the power subsystem for all operational spacecraft. Another FOT member might want to monitor a specific instrument which is flying aboard several spacecraft simultaneously. Still another FOT member might want to examine from a high level the current state of all the spacecraft and their instrument payloads. The FOS must be able to simultaneously have all the data available for all the spacecraft and instruments and be able to format that data according to the demands of the FOT.

In addition to the command and control capabilities, the FOS must also provide off-line capabilities. These would include the planning and scheduling of the spacecraft and instrument resources, the scheduling of the communications network required for forward and return service to the spacecraft, the generation and maintenance of the spacecraft and instrument loads, routine performance analysis and trending, and assisting in the identification of anomalies and their resolution. The FOS must integrate the real-time and off-line capabilities into one cohesive, tightly-coupled system.

The performance and security requirements for an operational spacecraft are much different from those of an integration and test system. The FOS is required to report the loss of any data, regardless of how little it may be. The FOS must also provide a failover architecture to ensure continued processing in the event of hardware failure. The FOS must be operational well over 99% of the time with no interruption to the services provided. The FOS must ensure that only authorized personnel are able to send commands to the spacecraft and instruments. Additionally, the FOS must maintain the integrity of the database and only changes submitted through the configuration control board are applied. In summary, the FOS must be operational virtually all the time with no loss of data, provide an architecture to handle hardware failures, maintain the strictest control for command authority and database integrity, and provide the resources and applications to support the multi-mission environment of the EOS program.

The biggest concern when evaluating OASIS was how well it would be able to handle the complicated requirements of the FOS. The FOS is required to support a multi-spacecraft environment with each spacecraft and its payload potentially different. The FOS is currently required to support up to 7 spacecraft simultaneously. This would include 5 operational spacecraft, 1 spacecraft transitioning into operations, and 1 spacecraft in testing. Each spacecraft would contain a payload comprised of one or more instruments. The following subsections present the results of the evaluation with respect to the FOS. The results are grouped according to the evaluation criteria which were defined in Section 3.1.

4.2.1 Functional and Performance Requirements

As a result of the evaluation of OASIS, it has been determined that considering the complexity of the EOS mission, OASIS is unable to support the needs and requirements of the FOS as it currently exists. Too many inconsistencies have been found between the requirements of the FOS and OASIS's ability to support those requirements. Overall, OASIS was found to satisfy less than 15% of the total FOS requirements. Even when considering the real-time portion of the FOS requirements (i.e. command, telemetry, and user interface), for which OASIS was specifically designed, it was only able to support approximately 25% of the requirements. It should also be noted that the AM1 spacecraft provider, MMC, has reported that OASIS is only able to satisfy approximately 30% of their I&T requirements. MMC had anticipated OASIS being able to support around 70% of their requirements but supplemental software had to be developed by MMC to augment the areas in which OASIS was deficient. The following table summarizes the analysis of the FOS requirements with respect to the ability of OASIS to satisfy them:

Table 4-1. FOS/OASIS Requirements Analysis Summary

Subsystem	Requirements Supported	Requirements Not Supported	Could Not Determine
Planning & Scheduling	0%	100%	0%
Command Management	0%	93%	7%
Command	21%	70%	9%
Telemetry	27%	58%	16%
Analysis	23%	77%	0%
Data Management	28%	72%	0%
Resource Management	0%	92%	8%
User Interface	28%	58%	14%
Performance	0%	50%	50%

In many cases it has been determined that although OASIS may provide a specific function, the way that function has been implemented in OASIS is unacceptable. An example of this is an equation for calculating a pseudo-telemetry point. Within OASIS once the equation had been defined, OASIS would have to be stopped and restarted to get the equation into the system. This is totally unacceptable within the FOS requirements. This same procedure is also required for adding procedures, equations, triggers, and display page definitions to the system. The FOS would not be permitted to stop processing and restart every time a change needed to be added for any of these services as this would cause a loss of data and interrupt all processing being performed by the FOT.

OASIS does not support the full protocol for Consultative Committee for Space Data Systems (CCSDS). OASIS is only able to handle the data portion of the packet, it can not process the header information. This places severe limitations on OASIS because it is unable to discriminate between the different data streams which the FOS would be receiving. It also excludes OASIS from verifying that commands had been received by the spacecraft to determine if a retransmission were required.

Both the database and commanding functions within OASIS lack the security measures required by the FOS. Within OASIS anyone is able to modify the contents of the database and commit those changes permanently. OASIS provides the mechanism to allow a user this privilege. Although this feature is a requirement of an I&T system, the management of the database within an operational environment is strictly controlled. OASIS does require command authority to send a command, however this authority is controlled via the USER_PRIVELEGES table and the system does not prevent anyone from modifying this table to get command authority.

The telemetry subsystem in OASIS, although able to handle the I&T environment, lacks the sophistication required to support the needs of the FOS. The FOS must be developed to be able to handle a variety of types of telemetry processing. Although OASIS is able to support the routine decommutation process, it is not able to handle some of the more complex processing such as decommutation of disjoint telemetry parameters, proper decommutation of data which appears multiple times in a single block of raw telemetry, and determination of the proper decommutation algorithm based upon the value of a known key indicator. The engineering unit (EU) conversion process of OASIS is restricted to a 3^{rd} order polynomial. Historically many control centers have been required to handle up to 7^{th} order polynomial conversions. OASIS is only able to support one format of telemetry data at a time. If the format changed, OASIS would have to be stopped and the new database tables copied into the OASIS directories and then OASIS restarted. The FOS needs to be able to recognize a change in format and access the appropriate database information for processing that data without restarting the system. Additionally, context dependent limit checking and EU conversion are not supported by OASIS. The FOS can not rule out that this type of processing would be required and must be developed to handle this condition if need be.

The FOS is also required to support the processing of memory dump data. OASIS is unable to process the dump data if it is embedded within the housekeeping data or if it is received simultaneously with the housekeeping data. This is due to the fact that the External Communications subsystem strips the header information from the packets before passing the data to the Data Handling subsystem. The Data Handling subsystem would therefore not be able to differentiate between the housekeeping and the dump data. During the FOS operations it is quite possible that the dump data will be either embedded in the housekeeping data or will be received simultaneously with the real-time telemetry from the spacecraft, so therefore OASIS would not be able to support the processing of that dump data. For the AM1 mission it is known that the memory dump data will come down from the spacecraft on the diagnostic channel and will therefore be a separate stream into the FOS.

The time tagging and correlation of all data within the control center is a very important feature for ensuring the health and safety of the spacecraft and its payload. This is of particular importance when investigating and resolving performance anomalies and degradations. All data and activities need to be time tagged with a standardized, non-volatile time source. That source needs to be the spacecraft time. Once time tagged, spacecraft events and data can be correlated to yield invaluable data when investigating an anomaly or failure. OASIS is unable to recognize or calculate spacecraft time. OASIS uses the system time for all its time tagging needs which is very unreliable and very volatile.

OASIS, although it does support a mechanism for recording data, does not provide the capability to perform history logging and replay as is required of the FOS. The OASIS recording feature captures in a file the data from a single stream and time tags that data with the system time. It has also been found that OASIS does not support the recording of command data. If telemetry and events were to be recorded they would be placed in separate files with no mechanism for correlating the two data sources. If the data recording was turned off and turned back on, the data would then be written to a new file. Every time recording is enabled for a stream the data is written to a new file which is time tagged with the current system time. There is no way to merge back orbit data that would be retrieved from the spacecraft recorders with the data received in real-time because the data is recorded with the current system time, not the calculated spacecraft time. While there may not be a need to merge real-time and recorder data in an I&T environment, the FOS is required to provide this capability.

OASIS performance is another area of concern. It has been found that if OASIS is running on a workstation that is participating in a client/server relationship, and the OASIS workstation is acting as the client, OASIS may abort or even bring down the workstation. The FOS is required to maintain a CPU utilization level below 50% while simultaneously supporting planning and scheduling, monitoring, and commanding of the spacecraft. Although accurate benchmarks on the performance of OASIS are unavailable, experience has indicated that the use of equations and procedures produce a CPU loading problem. Since OASIS has placed a lot of its control in CSTOL rather than in the database, this requires the system to utilize procedures and equations to meet certain requirements. For example, all pre-requisite state checks and end item verification of command execution must be defined as procedures through CSTOL rather than within the database. This is both a burdensome task to enter this information and requires the stopping and restarting of OASIS every time a change is required to the system. These procedures along with the those for triggers and the CSTOL defined equations could potentially produce a severe CPU loading problem, particularly during periods of high commanding rates.

CSTOL has been found to be a very wordy language. It requires many keystrokes and does not accept abbreviations. CSTOL also does not support the concept of command mnemonics. A command mnemonic is a shortened English-like ASCII representation of the command string. Command mnemonics are typically used as a short hand for the command string and are useful for saving time and keystrokes.

4.2.2 Flight Operational Requirements

As has been previously stated, the FOS is required to support a multi-mission environment in which it must currently support up to 7 spacecraft simultaneously. Since the current operational concept is to provide a multi-user environment in which the FOT will have the capability to view any combination of spacecraft and instruments simultaneously, the ground system must be designed to support this feature. Currently OASIS is being utilized as a single user system in support of a single spacecraft or instrument. Although Release Notes report OASIS is able to support a multi-user environment, it was not tested due to the fact that only a single workstation was available for the evaluation. However, it has been determined that due to the way OASIS handles CCSDS packets, OASIS would not be able to support an environment in which multiple streams from multiple spacecraft were being received in the FOS. OASIS would be unable to differentiate between the streams. In addition OASIS is only able to access a single database at a time. In order to change to a new database, the software would need to be stopped and the appropriate database copied into the OASIS directories and the software restarted. For simultaneous support of multiple spacecraft, each of which has its own unique database, the ground system must be able to determine which database is required and access that database as needed without stopping and restarting the system.

It also appears that OASIS would not be able to support a scenario in which contact with the spacecraft was very limited. OASIS does not support the generation and storage of preplanned command groups to be used in contingency situations. It also does not support the generation and storage of command loads. OASIS does support the building of a load file, but as soon as the file is closed it is forwarded to the External Communications subsystem for transmission. OASIS is not able to store the load for later use. The load file generated by OASIS is only able to support absolute time commands, it can not support relative time commands, flight software updates, and table updates, all of which are required of the FOS. OASIS also does not provide any level of validation on the load file.

4.2.3 Integration into the FOS

The current concept for the FOS is that the real-time and the off-line processing be combined into a single integrated system. The FOS would then be able to provide the tools necessary to support all stages and requirements of the ground based control center. The FOS is currently being designed through an object oriented methodology and planned implementation with the C++ programming language. The design of the FOS must be such that the system be developed as a single system from the ground up. It should not be developed as separate individual components which are then pieced together as best as is possible. OASIS does not provide many of the off-line activities such as planning and scheduling, spacecraft/instrument analysis, data archiving and retrieval, and command load generation. OASIS was developed as a complete system, it was never developed with the goal that it would be integrated into another larger system. It therefore is not clear how or even if OASIS could be integrated with the off-line components of the FOS to produce an integrated system capable of supporting the defined requirements.

4.2.4 Additional Functionality/Evolvability

The FOS is required within the Statement of Work (SOW) to "develop a design strategy to create a system that can gracefully grow and evolve, is flexible, and has reduced risk, complexity, and cost."

An assessment of the ability of OASIS to accept new functionality or its ability to evolve with changes to the scope of the work was not possible during this evaluation. It is known that the current process for requesting enhancements or corrections to OASIS deficiencies is through the OASIS Software Support/Change Request (OSCAR). Once an OSCAR is submitted it is assigned a number and priority for tracking. Due to the fact that the detailed design of OASIS was not known, an evaluation of the ability of OASIS to support added functionality was not possible. It does appear however that any additional functionality which would be desired of OASIS would have to be requested via an OSCAR. The changes would then be made by LASP and scheduled for delivery in a future release of OASIS.

4.2.5 Changes in Industry Standards

The SOW specifies that the FOS will use commercial hardware and software wherever it is cost effective. It also specifies that applications must be able to migrate from hardware platform to hardware platform without having to rewrite the applications or perform manual conversions. Additionally the SOW requires that the FOS be able to accept upgraded commercial off-the-shelf (COTS) software packages without requiring major changes to the application software. The current version of OASIS does not meet the requirements specified in the SOW. Currently OASIS can only be run on a Sun Workstation. It is not transportable to other comparable hardware platforms. OASIS utilizes TAE+ for building the user interface. OASIS does not accept the industry standard version of TAE+, it requires customization to TAE+. Any upgrades which would be planned for implementation into OASIS would require additional customization of TAE+ to run in the OASIS environment.

4.3 Summary

To summarize, OASIS has been evaluated and found to be a valuable tool when used within its limitations. OASIS is well suited for an I&T environment and is capable of performing well within that environment. However, in an operational environment in which there are many requirements above and beyond the traditional telemetry and command capabilities, OASIS has been found to be deficient. Indications also suggest that OASIS could not be easily integrated with the FOS off-line systems to produce a cohesive, tightly coupled system. OASIS has several nice features which need to be considered for use in the FOS, but the use of OASIS as a total package is not recommended.

5. Conclusions

Due to the fact that the AM1 and PM1 projects have chosen OASIS as their I&T tool, there has been some concern raised regarding the use of OASIS within the FOS. This is a very valid concern. The FOS is sensitive to this and initiated this study to evaluate the possible use of OASIS within the FOS. The results of the evaluation revealed that although OASIS provides the framework for the integration and testing of the spacecraft and instruments, it is not suitable for the operational FOS and its complex architecture.

The desire to possibly use OASIS in the FOS is a result of the spacecraft and instrument engineers having developed their own customized displays and procedures for integrating and testing of their instruments. They have become familiar with their displays and know where and what to look for when evaluating the performance. They have developed procedures which perform their specialized tasks. The real concern here is with the overall look and feel of the system, not over the underlying code which executes.

5.1 Recommendations

It is the recommendation of this study is to exploit those features which were found to be innovative within OASIS and have the FOS investigate the feasibility of incorporating them into the FOS design. In particular, the trigger and trend limit features should be considered for the FOS. It is also recommended that the FOS provide the tools to allow the spacecraft and instrument providers the capability to create "OASIS like" displays and procedures. The FOS development team should be provided with a copy of the OASIS software to facilitate the possible use of these OASIS features within the FOS.

The FOS development team has already begun its analysis for the FOS user interface. This analysis includes the evaluation of existing systems, the evaluation of COTS products, surveys to users, detailed requirements analysis, and prototype development. Within the framework of this analysis, the FOS user interface group, along with the other user interface personnel on the ECS contract, evaluated several GUI builders, including TAE+. Builder Xcessory (BX) was chosen over TAE+ (and other products) for the following reasons:

- BX generates pure Motif code. TAE+ requires an extra layer of proprietary libraries. This unnecessary level of abstraction produces slower executables over pure Motif code.
- BX provides full access to Motif widgets within the tool. TAE+ provides access only to a subset of Motif widgets. Access to all of Motif from TAE+ is possible, but it requires entering by hand the Motif code, something that is done by BX.

Part of the FOS user interface will be a display builder. The display builder will allow spacecraft and instrument personnel the ability to create telemetry displays on the fly and include them into the system. The display builder will allow for creation of the same types of displays currently being used within OASIS. Once a display is built it can be stored in the database for future use by the builder or by anyone else.

The procedures developed for use in OASIS are defined through the CSTOL interface. The FOS will also provide a STOL-like interface. The exact design of that interface is yet to be determined but it is planned to allow for the definition of procedures as is currently available with OASIS and CSTOL. Although the syntax may vary from what OASIS uses, the ability to define a procedure to produce the same result will be provided. As is the case with display definitions, once procedures have been developed they too may also be stored in the database to allow for use whenever required.

Abbreviations and Acronyms

BX	Builder Xcessory
CCSDS	Consultative Committee for Space Data Systems
CLP	control language processor
COTS	commercial off-the-shelf
CSTOL	Colorado System Test and Operations Language
ECS	EOSDIS Core System
EOC	EOS Operations Center
EOS	Earth Observing System
FOS	Flight Operations Segment
FOT	Flight Operations Team
GSFC	Goddard Space Flight Center
I&T	integration and test
LASP	Laboratory for Atmospheric and Space Physics
MMC	Martin Marietta Corporation
NASA	National Aeronautics and Space Administration
OASIS	Operations and Science Instrument Support
SOW	statement of work
TAE	Transportable Applications Environment
TDM	time division multiplexed